Scientific Journal Impact Factor (SJIF): 1.711

e-ISSN: 2349-9745
p-ISSN: 2393-8161

International Journal of Modern Trends in Engineering and Research
www.ijmter.com

Design of ABE for flexible access control and efficiently sharing large files using Hadoop over the cloud

H B Shinde[1] and V S Nandedkar[2]

[1] PG Student Department of Computer Engineering, Padmabhooshan Vasantdada Patil Institute of Technology, Bavdhan, Pune
[2] Associate Professor Department of Computer Engineering, Padmabhooshan Vasantdada Patil Institute of Technology, Bavdhan, Pune

Abstract: Cloud computing is the internet based technology widely used in IT industry. Cloud provides everything as a service over internet such as IaaS, PaaS, and SaaS. There are various security and privacy issues on outsourced data. Along with Data confidentiality fine grained access control is strongly desired. There are various access control schemes employing Attribute based encryption have been proposed; many of them suffer from inflexibility in implementing complex access control policies. Attribute based encryption schemes provide the fine grained access to cloud data but computational cost of encryption is increases with complex access policies. In order to alleviate the problems in existing system we proposes the scheme that will use the Hadoop framework which uses the mapreduce technique in order to process all user requests in parallel. In Hadoop HDFS process and stored large amount of data efficiently on distributed environment. we also efficiently revoked the user who will not be able to access to data which he already had access to. Hadoop is a software framework comes with a distributed file system called HDFS (Hadoop Distributed File System) which automatically handles many problems such as task scheduling, fault tolerance and inter-machine communication. Use of HDFS helps in storing the large amount of data on cloud in short time. Data stored in HDFS can be processed using Map Reduce programming model which comes with Hadoop.

Technical Keywords : Cloud**,** Attribute based encryption, CP-ABE, HDFS, mapreduce

I. INTRODUCTION

Cloud computing is an internet based technology which is built on virtualization, analogous and distributed computing. There are Different service-oriented cloud computing models have been implemented such as Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and Software as a Service (SaaS). Lastly in several years, cloud computing has evolve as one of the most prominent paradigms in the IT industry. Cloud computing provides many benefits to IT companies, academician and cloud users. Though the cloud provides various benefits there are some security issues that will prevent cloud computing extensive application and usage in future. To achieve the data confidentiality data must be encrypted before the uploaded to cloud. Traditional public key infrastructure can be used in data encryption process but there is problem that data owner should know the public key of every Data user and storage overhead is also increases. To alleviate this problem Sahai and waters proposed attribute based encryption [7]. In Attribute based encryption system encrypted data restrict the access to the authorized users who have matching attributes. There are two approaches for the Attribute based encryption CP-ABE and KP-ABE. In CP-ABE[6] scheme cipher text is associated with access policy and secret key is associated with the set of attributes and the only users with secret key which satisfies the access policy can able to decrypt the cipher text. In

KP-ABE [4] scenario is reverse i.e. user secret key is associated with access policy and cipher text is associated with set of attributes. In hierarchical attribute based encryption [1] users are organized as per their hierarchical structure. It uses the CP-ASBE [5] form where the recursive attribute set structure is used to describe the policy and it uses the conjunctive as well as disjunctive normal form to design the flexible access policies over recursive set structure. We propose the system that uses the cipher text policy ABE which is implemented over Hadoop framework in order to increase the efficiency and we can upload large file size data using HDFS and mapreduce [2]

II. LITERATURE SURVEY

Many schemes have been proposed for the flexible and fine grained access control [9]-[12] but the drawback of these schemes is that both data owner and service provider must be in same trusted domain. In cloud environment data owner and service provider will not be in the same trusted domain. New access control schemes considering ABE [7] has been proposed.

2.1 Attribute based Encryption

Sahai and Waters [9] have proposed ABE as new method for identity based encryption. In ABE plain text is encrypted with set of attributes. The Key generation server provides decryption key to the user which is associated with set of attribute. User can decrypt the cipher text if and only if there is match between attributes of ciphertext and users decryption key. Drawback of ABE is that it required efficient key management technique for distributing the keys to authorized users only which is difficult and also it lacks scalability and flexibility ; as number of users increases.

2.2 Key policy Attribute based Encryption (KP-ABE)

In KP-ABE [4], decryption key is associated with access structure and cipher text is associated with set of attributes. User is able to decrypt the ciphertext only if the attributes associated with ciphertext satisfy the tree access structure available with decryption key. The problem with this scheme is that data owner only chooses attributes for data and is not able to decide who can decrypt the encrypted data.

2.3 Ciphertext policy Attribute based Encryption (CP-ABE)

In CP-ABE [6] ciphertext is associated with access structure which is constructed using policy and decryption key is associated with set of attributes. CP-ABE is more closer to traditional Role-Based Access control [6]. Basic CP-ABE [6] unable to provide better flexibility and efficiency in specifying policies to manage user attributes [5]. Drawback of CP-ABE scheme is that decryption keys only support user attributes organized in single set so user uses all possible combination of attributes from given attribute set. In order to solve this problem Bobba [5] proposed ciphertext policy Attribute set based encryption (CP-ASBE or ASBE). ABSE is extension to CP-ABE which organizes the users attributes into recursive set structure. With ASBE we can impose dynamic constraints for the combination of attributes from different set to satisfy access policy. Wang [3] proposed hierarchical attribute based encryption (HABE) to achieve fine grained access in cloud storage by combination of hierarchical Identity based encryption (HIBE) and CP-ABE

III. PROPOSED SYSTEM

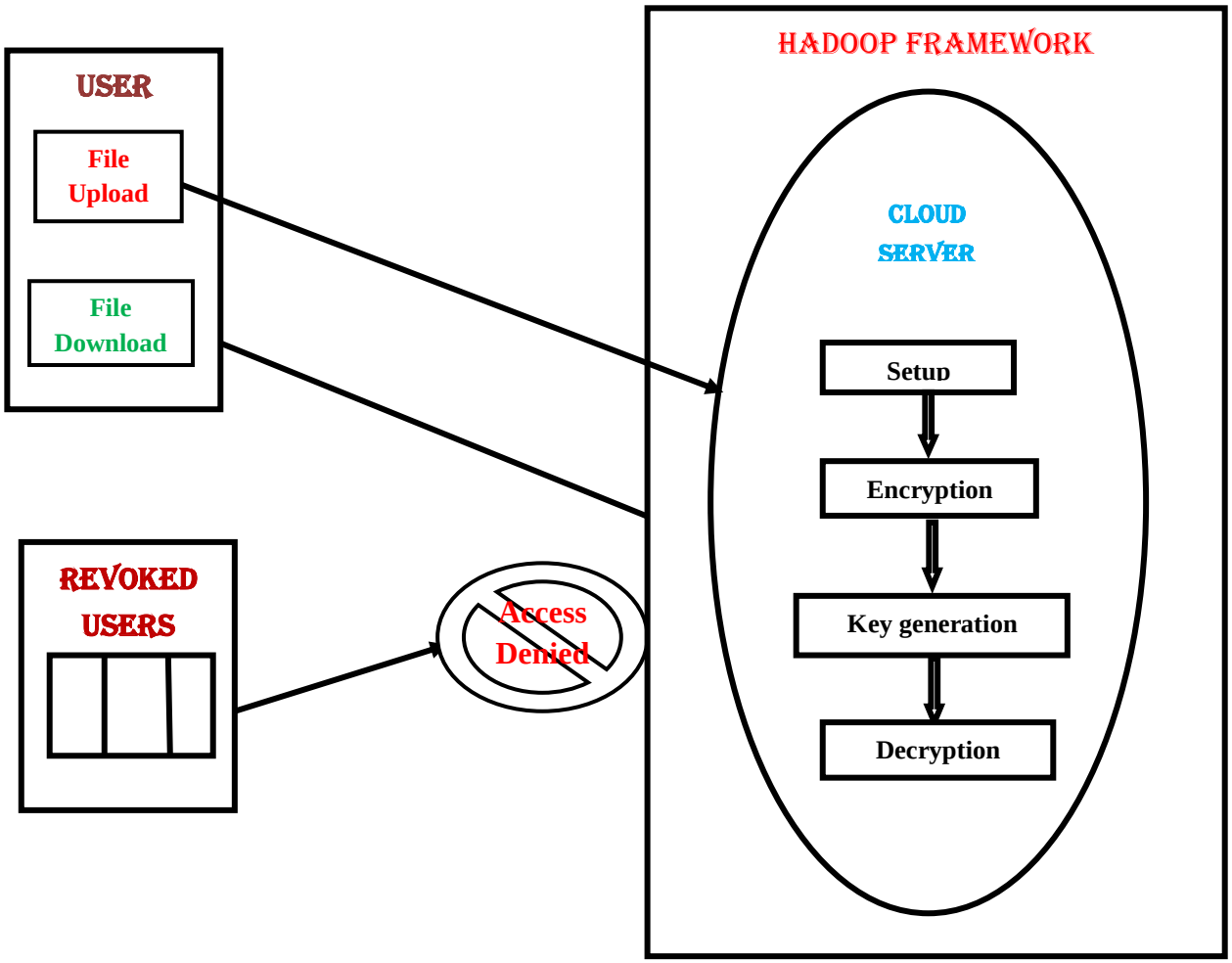

Fig 3.1 General Architecture of Proposed System

Figure 3.1 shows the general architecture of system which consist of five parties such as cloud service provider, data owner, data consumer, Domain authority and revoked user. System is implemented using Hadoop framework which has the HDFS and MapReduce tools used for programming purpose. Hadoop is used in order to process every user request in parallel so no user request is kept into waiting queue.

Data storage service is provided by Cloud Service Provider (CSP) to data owner. Data owners encrypt their data files and store it on the cloud for sharing with data consumers. To access the shared data files, data consumers download encrypted data files of their interest from the cloud and then decrypt them. Data owner/consumer is controlled by a domain authority called as Administrator. User can be Data owner or Data consumer or both. Setup algorithm is executed very first which will create the Public key and Master Secret key. This public key is used by Data Owner to encrypt the files before storing on cloud and set the access policy which describe who can able be decrypt this file. When Data consumer want to access the files stored on cloud, Key Generation algorithm is executed by server and the secrete key for the user is created depending upon the attribute set user is possessing. Any user is able to access the encrypted files from the server if and only if there is match between the secrete key attributes and access policy attributes otherwise user get the NULL value. The system is implemented using Hadoop framework so that user can upload or

download the large files in short time and the entire task will be done in parallel. The revoked users are not allowed to have access to the server data.

IV. ALGORITHMIC STRATEGY

The algorithms of proposed system have the five sub algorithms which are

Setup: - This setup algorithm is executed by the Root authority of system and it generate the public key and master key which are used for the further processing of Attribute based encryption

Setup algorithm	
Input: Security parameter K, Universe attribute set U	
Step 1	Consider the two bilinear cyclic group G1,G2
Step 2	Randomly choose the two numbers P € G1 and Q € G2 (P and Q are the generating point of group G1 and G2 respectively)
Step 3	Choose the random numbers α and β from Z_p
Step 4	select the hash function H which maps the every element from the attribute set to group element
Step 5	Generate the Public key and Master key
Output: public key and master key	

Encryption:- In this step user can encrypt their file before storing it on cloud storage. Public key is used to encrypt the file. Output of this algorithm is encrypted file or cipher text. This algorithm is taking the Public key ,file to be uploaded and access tree as input and produce the encrypted file which is to be stored on server.

Key Generation:-This algorithm is used to create the Secrete key for every user. The input required for the algorithm is Master key generated during setup and users attribute set A. some randomized exponential operations are carried out and secrete key is generated for user U.

Decryption:-This algorithm is used to decrypt the file. This algorithm takes cipher text or encrypted file and secrete key as input and produce the plain text or decrypt the file. Any user is able to decrypt the file if and only if the attributes associated with every users secrete key satisfies the access policy set at the time of upload of encrypted file.

User Revocation:- Users in an organization are sharing the data. When user entered into system then he/she receives set of attributes and various keys required to upload and download the data. Any user is part of system till he is working with that organization when he left the job or resigned from the job he will no longer be the part of that organization so he should be revoked and does not allowed to access the system data anymore. So efficient revocation mechanism is desired in every system in order to prevent the data to be used by unauthorized users.

V. MATHEMATICAL MODEL

During the execution of proposed algorithm there are some mathematical operations that are carried out and we are getting some output.

Generating Public key and Master key: Execution of setup algorithm provide the public key and master key which are used in encrypting file and generating the users secrete key.

$$Pub\_key=(G_0, g, h=g^{\beta}, f=g^{1/\beta}, e(g,g)^{\alpha}) \ldots\ldots\ldots\ldots 1$$

$$Matser\ Key= (\beta, g^{\alpha}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 2$$

Where,
G_0 is bilinear cyclic group
g be generator of G
α,β are random number from set Zp
h,f are hash function which maps the attribute element into group element

Generating the users secrete key: During the key generation algorithm users secrete key is generated as shown below

$$SK= g^{(\alpha+s)/\beta} \text{ for all } j \in A \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3$$

Where,
A is attribute set of user
α,β are random number from set Zp
s is random number from Zp
j is attribute from users attribute set

VI. DATA SETS AND EXPECTED RESULTS

6.1 Data set:

Data set for the system should contains any kind of file such as it can be Pdf or doc, text file or any other file.

6.2 Expected Results:

We are using the Hadoop for implementation of attribute based encryption for flexible and scalable access control. In traditional Attribute based encryption computational cost of encryption process is increasing due to complex access structure. In Hadoop we have map reduce and HDFS tools which will be used to store and process the large amount of data very fastly. So in proposed system the amount of time required to encrypt and upload the file will be lesser than existing scheme. We will achieve the data confidentiality and flexible access control over the cloud data and efficient user revocation. Also we can share large file size data on cloud storage.

VII. CONCLUSION AND FUTURE ENHANCEMENT

Thus we have design the system which will achieve the data confidentiality and provide the fine grained and scalable access to data stored on cloud storage. Also the large amount of files can be uploaded and downloaded in short time period using HDFS. My contribution will be to use the MapReduce a programming tool to process large amount of data with the help of HDFS. And also efficiently revoked the users who will not get access to system. Our work can be further extended to implement this system on structured data such as relational database.

REFERENCES

[1] Zhiguo Wan, Jun'e Liu and Robert Deng, "Hierarchical attribute based solution for flexible and scalable access control in cloud computing" in Proc. IEEE transaction on information Forensics and Security, Vol.7 Singapore, 2012

[2] Tom White, "Hadoop: The Definitive Guide", First Edition, Published by O'Reilly Media, June 2009, in United States of America.ao,Tao Li , IEEE transaction,2012

[3] G.Wang, Q. Liu, and J.Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," in Proc. ACM Conf. Computer and Communications Security (ACM CCS), Chicago,IL, 2010

[4] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proc. IEEE INFOCOM 2010, 2010, pp. 534–542.

[5] R. Bobba, H. Khurana, and M. Prabhakaran, "Attribute-sets: A practically motivated enhancement to attribute-based encryption," in Proc. ESORICS, Saint Malo, France, 2009

[6] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute based encryption," in Proc. IEEE Symp. Security and Privacy, Oakland, CA, 2007

[7] V. Goyal, O. Pandey, A. Sahai, and B.Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. ACM Conf. Computer and Communications Security (ACM CCS), Alexandria,VA, 2006

[8] A. Sahai and B. Waters, "Fuzzy identity based encryption," in Proc. Advances in Cryptology—Eurocrypt, 2005, vol. 3494, LNCS, pp. 457–473.

[9] J. Li, N. Li, and W. H. Winsborough, "Automated trust negotiation using cryptographic credentials," in Proc. ACM Conf. Computer and Communications Security (CCS), Alexandria, VA, 2005.

[10] T. Yu and M. Winslett, "A unified scheme for resource protection in automated trust negotiation," in Proc. IEEE Symp. Security and Privacy, Berkeley, CA, 2003

[11] P. D. McDaniel and A. Prakash, "Methods and limitations of security policy reconciliation," in Proc. IEEE Symp. Security and Privacy, Berkeley, CA, 2002

[12] H. Harney, A. Colgrove, and P. D. McDaniel, "Principles of policy in secure groups," in Proc. NDSS, San Diego, CA, 2001

